



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년08월25일
(11) 등록번호 10-1059658
(24) 등록일자 2011년08월19일

(51) Int. Cl.

G06F 9/44 (2006.01) G06F 13/36 (2006.01)

G06F 13/38 (2006.01)

(21) 출원번호 10-2010-0063582

(22) 출원일자 2010년07월01일

심사청구일자 2010년07월01일

(56) 선행기술조사문헌

US2006/0230396 A1

US2007/0261027 A1

US2005/0251809 A1

KR1020030021554 A

전체 청구항 수 : 총 15 항

(73) 특허권자

엔에이치엔(주)

경기 성남시 분당구 정자동 178-1 그린팩토리

(72) 발명자

이동수

경기도 용인시 기흥구 보정동 대덕누리에뜰아파트
101동 1203호

(74) 대리인

특허법인무한

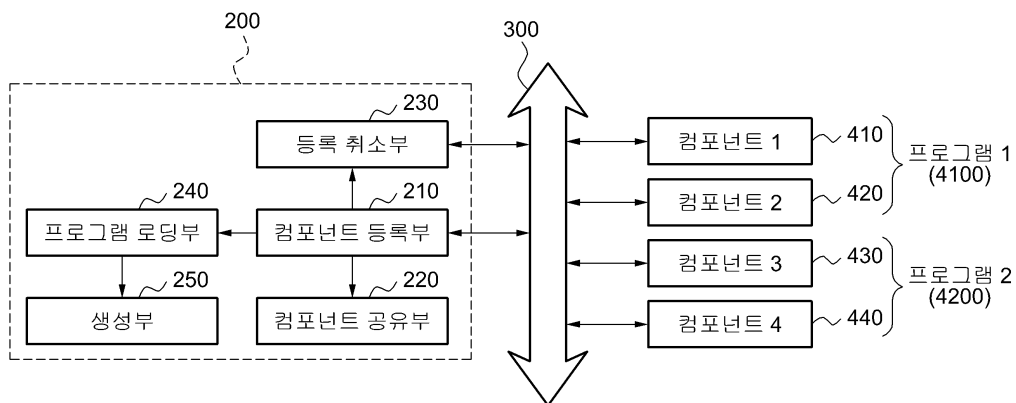
심사관 : 윤혜숙

(54) 개발자 인터페이스 제공 방법 및 시스템

(57) 요약

로딩된 프로그램을 이용하여 개발자 인터페이스를 생성하는 방법 및 시스템이 개시된다. 개발자 인터페이스 생성 방법은 로딩된 프로그램을 구성하는 복수의 컴포넌트들을 통신 인터페이스를 통해 공유하고, 공유되는 복수의 컴포넌트들 중 개발자 인터페이스를 통해 선택된 컴포넌트들을 이용하여 새로운 프로그램을 생성할 수 있다.

대표도 - 도2



특허청구의 범위

청구항 1

복수의 컴포넌트들(Component)로 구성된 프로그램 1을 로딩하는 단계; 및

상기 로딩된 프로그램 1을 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트를 이용하여 프로그램 3을 생성하는 단계

를 포함하는 개발자 인터페이스 제공 방법.

청구항 2

제1항에 있어서,

상기 로딩된 프로그램 1과 컴포넌트를 공유하는 하나 이상의 프로그램 2를 로딩하는 단계

를 더 포함하는 개발자 인터페이스 제공 방법.

청구항 3

제2항에 있어서,

상기 프로그램 3을 생성하는 단계는,

상기 프로그램 1을 구성하는 적어도 하나의 컴포넌트와 상기 프로그램 2를 구성하는 적어도 하나의 컴포넌트를 이용하여 상기 프로그램 3을 생성하는 것을 특징으로 하는 개발자 인터페이스 제공 방법.

청구항 4

제2항에 있어서,

상기 로딩한 프로그램 1 및 2를 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출하는 단계

를 더 포함하는 개발자 인터페이스 제공 방법.

청구항 5

제4항에 있어서,

상기 프로그램 3을 생성하는 단계는,

상기 개발자 인터페이스를 통해 노출된 프로그램 1 및 2를 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트 선택을 입력 받는 단계; 및

상기 선택을 입력 받은 적어도 하나의 컴포넌트들을 결합하여 상기 프로그램 3을 생성하는 단계

를 포함하는 개발자 인터페이스 제공 방법.

청구항 6

제4항에 있어서,

상기 개발자 인터페이스는, XML 기반의 동적 개발자 인터페이스인 것을 특징으로 하는 개발자 인터페이스 제공 방법.

청구항 7

제1항에 있어서,

상기 컴포넌트의 타입은,

동적 링크 라이브러리(Dynamic Linked Library: DLL) 및 실행 프로그램(Execute Program: EXE) 중 어느 하나인

것을 특징으로 하는 개발자 인터페이스 제공 방법.

청구항 8

제1항 내지 제7항 중 어느 한 항의 방법을 실행하기 위한 프로그램이 기록된 컴퓨터에서 판독 가능한 기록 매체.

청구항 9

복수의 컴포넌트들(Component)로 구성된 프로그램 1을 로딩하는 프로그램 로딩부; 및

상기 로딩된 프로그램 1을 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트를 이용하여 프로그램 3을 생성하는 생성부

를 포함하는 개발자 인터페이스 시스템.

청구항 10

제9항에 있어서,

상기 프로그램 로딩부는,

상기 로딩된 프로그램 1과 컴포넌트를 공유하는 하나 이상의 프로그램 2를 로딩하는 것을 특징으로 하는 개발자 인터페이스 시스템.

청구항 11

제10항에 있어서,

상기 생성부는,

상기 프로그램 1을 구성하는 적어도 하나의 컴포넌트와 상기 프로그램 2를 구성하는 적어도 하나의 컴포넌트를 이용하여 상기 프로그램 3을 생성하는 것을 특징으로 하는 개발자 인터페이스 시스템.

청구항 12

제10항에 있어서,

상기 로딩한 프로그램 1 및 2를 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출하는 인터페이스 노출부

를 더 포함하는 개발자 인터페이스 시스템.

청구항 13

제12항에 있어서,

상기 생성부는,

상기 개발자 인터페이스를 통해 노출된 프로그램 1 및 2를 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트 선택을 입력 받는 컴포넌트 선택부; 및

상기 선택을 입력 받은 적어도 하나의 컴포넌트들을 결합하여 상기 프로그램 3을 생성하는 프로그램 생성부

를 포함하는 개발자 인터페이스 시스템.

청구항 14

제12항에 있어서,

상기 개발자 인터페이스는, XML 기반의 동적 개발자 인터페이스인 것을 특징으로 하는 개발자 인터페이스 시스템.

청구항 15

제10항에 있어서,

상기 컴포넌트의 타입은,

동적 링크 라이브러리(Dynamic Linked Library: DLL) 및 실행 프로그램(Execute Program: EXE) 중 어느 하나인 것을 특징으로 하는 개발자 인터페이스 시스템.

명세서

기술분야

[0001] 본 발명은 개발자 인터페이스 제공 방법 및 시스템에 관한 것으로, 보다 구체적으로 커뮤니케이션 버스를 이용하여 등록된 컴포넌트들의 실행에 따른 개발자 인터페이스를 제공하는 방법 및 시스템에 관한 것이다.

배경기술

[0002] 소프트웨어 개발에 있어 매우 강조되는 한가지가 바로 재사용성이다. 개발이 완료된 소프트웨어를 다른 여러 소프트웨어에서 재사용하는 것은 추가개발이 필요없는 비용 감소, 이미 사용되고 있는데 서 오는 안정성 확보 등의 여러 가지 장점이 있다.

[0003] 그 동안 소프트웨어의 재사용은 코드 또는 컴파일 완료상태의 바이너리 재사용에 국한되어 왔다. 프로그램의 개발자는 다른 개발자가 만든 코드를 재사용하기 위해 소스파일을 가져와 자신의 프로그램과 함께 컴파일하거나 프로그램(exe, dll)을 재사용하기 위해서 이 파일들을 복사하여 자신의 프로그램에서 재사용하곤 했다.

[0004] 하지만, 필요에 따라 코드나 dll과 같은 개발이 완료된 코드를 넘어 이미 메모리에 로딩된 살아있는 다른 프로그램을 재사용한다면, 보다 나은 재 사용성을 제공할 수 있다.

[0005] 예를 들어, 메신저 프로그램이 동작하는 상태에서 게임 프로그램이 이 메신저 프로그램을 재사용한다면, 게임 프로그램이 메신저 프로그램안에 있는 로그인된 사용자 정보를 가져와 게임안에서 로그인된 사용자 정보를 활용하여 게임안에서 해당 사용자를 초대할 수도 있을 것이다.

발명의 내용

해결하려는 과제

[0006] 본 발명은 커뮤니케이션 버스를 이용하여 등록된 다수의 컴포넌트들 간의 데이터를 공유하는 방법 및 시스템을 제공한다.

[0007] 본 발명은 등록된 다수의 컴포넌트들을 실행하면서 신규 컴포넌트를 추가하는 방법 및 시스템을 제공한다.

[0008] 본 발명은 커뮤니케이션 버스를 이용하여 등록 중인 컴포넌트의 등록을 취소하는 방법 및 시스템을 제공한다.

[0009] 본 발명은 개발자 인터페이스를 활용하여 이미 로딩된 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 방법 및 시스템을 제공한다.

과제의 해결 수단

[0010] 본 발명의 일실시예에 따른 개발자 인터페이스 제공 방법은, 복수의 컴포넌트들(Components)로 구성된 프로그램 1을 로딩하는 단계, 및 상기 로딩된 프로그램 1을 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트를 이용하여 프로그램 3을 생성하는 단계를 포함할 수 있다.

[0011] 또한, 상기 로딩된 프로그램 1과 컴포넌트를 공유하는 하나 이상의 프로그램 2를 로딩하는 단계를 더 포함할 수 있다.

[0012] 또한, 상기 프로그램 3을 생성하는 단계는, 상기 프로그램 1을 구성하는 적어도 하나의 컴포넌트와 상기 프로그램 2를 구성하는 적어도 하나의 컴포넌트를 이용하여 상기 프로그램 3을 생성할 수 있다.

[0013] 또한, 상기 로딩한 프로그램 1 및 2를 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출하는 단계를 더 포함할 수 있다.

[0014] 또한, 상기 프로그램 3을 생성하는 단계는, 상기 개발자 인터페이스를 통해 노출된 프로그램 1 및 2를 구성하는

복수의 컴포넌트들 중 적어도 하나의 컴포넌트 선택을 입력 받는 단계, 및 상기 선택을 입력 받은 적어도 하나의 컴포넌트들을 결합하여 상기 프로그램 3을 생성하는 단계를 포함할 수 있다.

[0015] 이때, 상기 개발자 인터페이스는, XML 기반의 동적 개발자 인터페이스일 수 있다.

[0016] 또한, 상기 컴포넌트의 타입은, 동적 링크 라이브러리(Dynamic Linked Library: DLL) 및 실행 프로그램(Execute Program: EXE) 중 어느 하나일 수 있다.

[0017] 또한, 본 발명의 일실시예에 따른 개발자 인터페이스 시스템은, 복수의 컴포넌트들(Components)로 구성된 프로그램 1을 로딩하는 프로그램 로딩부, 및 상기 로딩된 프로그램 1을 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트를 이용하여 프로그램 3을 생성하는 생성부를 포함할 수 있다.

[0018] 또한, 상기 프로그램 로딩부는, 상기 로딩된 프로그램 1과 컴포넌트를 공유하는 하나 이상의 프로그램 2를 로딩할 수 있다.

[0019] 또한, 상기 생성부는, 상기 프로그램 1을 구성하는 적어도 하나의 컴포넌트와 상기 프로그램 2를 구성하는 적어도 하나의 컴포넌트를 이용하여 상기 프로그램 3을 생성할 수 있다.

[0020] 또한, 상기 로딩한 프로그램 1 및 2를 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출하는 인터페이스 노출부를 더 포함할 수 있다.

[0021] 또한, 상기 생성부는, 상기 개발자 인터페이스를 통해 노출된 프로그램 1 및 2를 구성하는 복수의 컴포넌트들 중 적어도 하나의 컴포넌트 선택을 입력 받는 컴포넌트 선택부, 및 상기 선택을 입력 받은 적어도 하나의 컴포넌트들을 결합하여 상기 프로그램 3을 생성하는 프로그램 생성부를 포함할 수 있다.

발명의 효과

[0022] 본 발명의 일실시예에 따르면, 커뮤니케이션 버스를 이용하여 등록된 다수의 컴포넌트들 간의 데이터를 공유할 수 있다.

[0023] 본 발명의 일실시예에 따르면, 등록된 다수의 컴포넌트들을 실행하면서 신규 컴포넌트를 추가할 수 있다.

[0024] 본 발명의 일실시예에 따르면, 커뮤니케이션 버스를 이용하여 등록 중인 컴포넌트의 등록을 취소할 수 있다.

[0025] 본 발명의 일실시예에 따르면, 커뮤니케이션 버스를 이용하여 등록 중인 컴포넌트를 조정할 수 있다.

도면의 간단한 설명

[0026] 도 1은 본 발명의 일실시예에 따른 커뮤니케이션 버스를 이용하여 컴포넌트를 등록하는 과정을 설명하기 위해 제공되는 도면이다.

도 2는 본 발명의 일실시예에 따른 본 발명의 일실시예에 따른 개발자 인터페이스 시스템의 구성을 도시한 도면이다.

도 3은 본 발명의 일실시예에 따른 개발자 인터페이스 시스템에서 생성부의 세부 구성을 도시한 도면이다.

도 4는 본 발명의 일실시예에 따라 로딩된 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 방법을 설명하기 위해 제공되는 흐름도이다.

도 5는 본 발명의 일실시예에 따라 커뮤니케이션 버스를 통해 공유되는 복수의 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 방법을 설명하기 위해 제공되는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0027] 이하, 본 발명의 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.

[0028] 도 1은 본 발명의 일실시예에 따른 커뮤니케이션 버스를 이용하여 컴포넌트를 등록하는 과정을 설명하기 위해 제공되는 도면이다.

[0029] 도 1에서, 컴포넌트 1(30), 컴포넌트 2(40), 및 컴포넌트 4(60)는 커뮤니케이션 버스(20)를 이용하여 이미 등록된 상태이고, 컴포넌트 3(50)을 등록하려는 경우, 개발자 인터페이스 시스템(10)은 커뮤니케이션 버스(20)를 이용하여 컴포넌트 3(50)을 등록할 수 있다.

- [0030] 이때, 개발자 인터페이스 시스템(10)은 컴포넌트 3(50)이 신규로 등록되었음을 컴포넌트 1(30), 컴포넌트 2(40), 및 컴포넌트 4(60)에 알릴 수 있다. 그러면, 커뮤니케이션 버스(20)를 통해 이미 등록된 컴포넌트 1(30), 컴포넌트 2(40), 및 컴포넌트 4(60)와 신규로 등록된 컴포넌트 3(50)가 서로 공유될 수 있다. 일례로, 컴포넌트 1, 2, 4와 신규로 등록된 컴포넌트 3(50)을 구성하는 데이터들이 커뮤니케이션 버스(20)를 통해 상호 공유될 수 있다. 여기서, 컴포넌트 1 내지 컴포넌트 4(30~60)를 구성하는 데이터는 XML 형식의 데이터일 수 있다. 이때, XML 형식의 데이터는 DTD(Document Type Definition)를 이용하여 정의될 수 있다.
- [0031] 도 2는 본 발명의 일실시예에 따른 개발자 인터페이스 시스템의 구성을 도시한 도면이다.
- [0032] 도 2를 참고하면, 개발자 인터페이스 시스템(200)은 컴포넌트 등록부(210), 컴포넌트 공유부(220), 등록 취소부(230), 프로그램 로딩부(240), 및 생성부(250)를 포함할 수 있다.
- [0033] 컴포넌트 등록부(210)는 커뮤니케이션 버스(Communication Bus: 300)를 이용하여 하나 이상의 컴포넌트를 등록할 수 있다. 여기서, 컴포넌트는 클라이언트에서 동작하는 다수의 서비스에 해당하는 프로그램을 구성하는 모듈이다. 즉, 프로그램은 복수의 컴포넌트들로 구성될 수 있다. 이때, 프로그램은 게임, 메신저, 블로그 등을 포함하고, 컴포넌트(component)는 게임, 메신저, 블로그 등의 프로그램을 구성하기 위한 웹 페이지, 아바타, 배경, 텍스트, 음악, 및 게임 아이템 등의 모듈을 포함할 수 있다.
- [0034] 일례로, 컴포넌트 1(410) 및 컴포넌트 2(420)는 프로그램 1을 구성하는 모듈이고, 컴포넌트 3(430) 및 컴포넌트 4(440)는 프로그램 2를 구성하는 모듈일 수 있다. 이때, 컴포넌트 1(410), 및 컴포넌트 2(420)는 기등록되고, 컴포넌트 3(430)을 신규로 등록하는 경우, 컴포넌트 등록부(210)는 커뮤니케이션 버스(300)를 이용하여 컴포넌트 3(430)을 신규로 등록할 수 있다. 그리고, 컴포넌트 등록부(210)는 컴포넌트 3(430)이 신규로 등록되었음을 커뮤니케이션 버스(300)를 이용하여 컴포넌트 1(410), 컴포넌트 2(420)에 알릴 수 있다.
- [0035] 또한, 컴포넌트 등록부(210)는 등록된 적어도 하나의 컴포넌트가 실행 중인 경우에, 신규 컴포넌트를 추가할 수도 있다.
- [0036] 그러면, 컴포넌트 공유부(220)는 커뮤니케이션 버스(300)를 이용하여 등록된 컴포넌트들을 서로 공유할 수 있다. 일례로, 컴포넌트 공유부(220)는 등록된 컴포넌트들을 구성하는 XML 형식의 데이터를 커뮤니케이션 버스(300)를 통해 서로 공유할 수 있다. 도 2에 따르면, 컴포넌트 공유부(220)는 하나의 런처(Launcher)에서 컴포넌트 1(410) 내지 컴포넌트 4(440)이 갖고 있는 데이터들을 커뮤니케이션 버스(300)를 이용하여 서로 공유할 수 있다. 이를 위해, 컴포넌트 공유부(220)는 등록된 컴포넌트로부터 수신한 데이터를 커뮤니케이션 버스(300)를 통해 다른 컴포넌트에 전송할 수 있다.
- [0037] 일례로, 게임인 프로그램 1(4100)을 구성하는 컴포넌트 1(410) 및 컴포넌트 2(420)와 메신저(messenger)인 프로그램 2(4200)를 구성하는 컴포넌트 3(430) 및 컴포넌트 4(440)가 커뮤니케이션 버스(300)를 통해 공유되는 경우, 개발자 인터페이스 시스템(200)은 메신저에 사용자가 로그인하였음을 게임 상에 표시할 수 있다. 이때, 컴포넌트 3(430)은 메신저 로그 정보에 해당하는 모듈이고, 컴포넌트 4(440)는 메신저 이모티콘, 글꼴에 해당하는 디자인 모듈인 경우, 컴포넌트 3(430)은 메신저에 등록된 사용자들이 메신저에 로그인할 때마다 메신저 로그 정보를 커뮤니케이션 버스(300)에 브로드캐스팅(broadcasting)할 수 있다. 이에 따라, 컴포넌트 3(430)은 사용자가 메신저에 로그인할 때마다 메신저 정보를 컴포넌트 1(410) 및 컴포넌트 2(420)로 전달하기 위해 컴포넌트 3을 수정할 필요가 없다.
- [0038] 그리고, 컴포넌트 공유부(220)는 커뮤니케이션 버스(300)를 통해 수신된 메신저 로그 정보를 커뮤니케이션 버스(300)를 이용하여 컴포넌트 1(410) 및 컴포넌트 2(420)로 전달할 수 있다. 그러면, 게임 상에 메신저에 사용자가 로그인한 정보가 표시될 수 있다. 이에 따라, 컴포넌트 1(410) 및 컴포넌트 2(420)는 제공되는 라이브러리를 그대로 사용하여 메신저 정보를 수신할 수 있다. 즉, 컴포넌트 1(410) 및 컴포넌트 2(420)로 구성된 프로그램 1(4100)은 메신저 로그 정보를 수신하기 위한 프로토콜을 설계하거나, 코드를 개발하지 않아도 된다.
- [0039] 다른 예로, 게임인 프로그램 1(4100)이 컴포넌트 1(410) 및 컴포넌트 2(420)로 구성되고, 블로그인 프로그램 3과 컴포넌트들을 공유하는 경우, 컴포넌트 1(410)은 동작하고 있는 게임의 현재 화면을 캡처(Capture)하여 커뮤니케이션 버스(300)로 전달할 수 있다. 그러면, 컴포넌트 공유부(220)는 수신된 캡처 화면을 커뮤니케이션 버스(300)를 이용하여 프로그램 3을 구성하는 컴포넌트들로 전달할 수 있다. 이에 따라, 블로그에는 캡처 화면이 업로드될 수 있다.
- [0040] 또 다른 예로, 게임인 프로그램 1(4100)과 웹 페이지(Webpage)인 프로그램 4를 구성하는 컴포넌트들이 공유되는

경우, 프로그램 1(4100)을 구성하는 컴포넌트 1(410)은 퀘스트(Quest) 수행과 관련된 웹 페이지 요청 데이터를 커뮤니케이션 버스(300)로 전달할 수 있다. 그러면, 컴포넌트 공유부(220)는 커뮤니케이션 버스(300)를 통해 웹 페이지 요청 데이터를 수신하고, 수신한 웹 페이지 요청 데이터를 커뮤니케이션 버스(300)를 이용하여 프로그램 4를 구성하는 컴포넌트들로 전달할 수 있다. 그리고, 컴포넌트 공유부(220)는 프로그램 4에서 커뮤니케이션 버스(300)를 통해 수신된 웹 페이지를 커뮤니케이션 버스(300)를 이용하여 프로그램 1(4100)을 구성하는 컴포넌트 1(410) 및 컴포넌트(420) 중 적어도 하나로 전달할 수 있다. 그러면, 게임 상에 퀘스트 수행과 관련된 웹 페이지가 표시될 수 있다.

[0041] 등록 취소부(230)는 신규 컴포넌트를 등록하는 중에 신규 컴포넌트의 등록 취소가 요청되면, 신규 컴포넌트의 등록을 취소할 수 있다. 이때, 신규 컴포넌트의 등록이 취소되더라도, 실행되고 있는 컴포넌트들은 실행 종료 없이 계속 실행될 수 있다.

[0042] 프로그램 로딩부(240)는 등록된 컴포넌트들을 구성하는 프로그램들을 로딩>Loading>할 수 있다. 이때, 프로그램 로딩부(240)는 등록된 컴포넌트들을 구성하는 프로그램들 중 어느 하나의 프로그램을 로딩한 상태에서 또 다른 등록 컴포넌트를 구성하는 프로그램을 로딩할 수도 있다. 여기서, 등록된 컴포넌트들은 커뮤니케이션 버스(300)를 통해 공유되므로, 프로그램 로딩부(240)는 커뮤니케이션 버스(300)를 통해 제1 프로그램(4100)과 컴포넌트들을 공유하는 하나 이상의 프로그램을 로딩할 수 있다.

[0043] 일례로, 프로그램 로딩부(240)는 게임인 프로그램 1(4100)을 로딩한 상태에서, 메신저인 프로그램 2(4200)를 로딩할 수 있다. 이때, 프로그램 1(4100)이 메인(main)으로 실행되는 상태에서 프로그램 2(4200)는 프로그램 1(4100)에 종속되어 실행될 수 있다. 즉, 프로그램 1을 메인으로 하는 하나의 런처(Launcher)에서 복수의 프로그램들이 실행될 수 있다.

[0044] 생성부(250)는 로딩된 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성할 수 있다. 일례로, 프로그램 1(4100)이 로딩된 경우, 생성부(250)는 프로그램 1(4100)을 구성하는 컴포넌트 1(410) 및 컴포넌트 2(420) 중 적어도 하나를 이용하여 프로그램 3을 생성할 수 있다.

[0045] 또한, 복수의 프로그램이 로딩된 경우, 생성부(250)는 복수의 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성할 수도 있다.

[0046] 일례로, 프로그램 1(4100) 및 프로그램 2(4200)이 로딩된 경우, 생성부(250)는 프로그램 1(4100)을 구성하는 복수의 컴포넌트들 및 프로그램 2(4200)를 구성하는 복수의 컴포넌트들 중 하나 이상의 컴포넌트를 선택할 수 있다. 그리고, 생성부(250)는 선택한 컴포넌트들을 결합하여 새로운 프로그램 3을 생성할 수 있다.

[0047] 이때, 생성부(250)는 프로그램 3을 생성하는 데 필요한 하나 이상의 신규 컴포넌트를 생성할 수도 있다. 그러면, 생성부(250)는 생성한 신규 컴포넌트와 선택한 컴포넌트들 결합하여 프로그램 3을 생성할 수도 있다.

[0048] 이하에서는 도 3을 참조하여, 로딩된 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 과정에 대해 상세히 설명하기로 한다.

[0049] 도 3은 본 발명의 일실시예에 따른 개발자 인터페이스 시스템에서 생성부의 세부 구성을 도시한 도면이다.

[0050] 도 3에 따르면, 생성부(250)는 인터페이스 노출부(251), 컴포넌트 선택부(252), 프로그램 생성부(253)를 포함할 수 있다.

[0051] 인터페이스 노출부(251)는 로딩된 하나 이상의 프로그램을 구성하는 컴포넌트들을 개발자 인터페이스(Developer Interface)를 통해 개발자에게 노출할 수 있다. 이때, 프로그램을 구성하는 컴포넌트의 타입(Type)은 동적 링크 라이브러리(Dynamic Linked Library: DLL), 및 실행 프로그램(Execute Program: EXE) 중 어느 하나일 수 있다.

[0052] 일례로, 프로그램 로딩부(240)에서 프로그램 1(4100)을 로딩한 경우, 인터페이스 노출부(251)는 프로그램 1(4100)을 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출할 수 있다. 즉, 인터페이스 노출부(251)는 개발자 인터페이스를 통해 컴포넌트 1(410) 및 컴포넌트 2(420)를 노출할 수 있다.

[0053] 다른 예로, 프로그램 로딩부(240)에서 프로그램 1 및 프로그램 2를 로딩한 경우, 인터페이스 노출부(251)는 프로그램 1(4100) 및 프로그램 2(4200)을 구성하는 복수의 컴포넌트들을 개발자 인터페이스를 통해 노출할 수 있다. 즉, 인터페이스 노출부(251)는 개발자 인터페이스를 통해 컴포넌트 1(410) 내지 컴포넌트 4(440)를 개발자에게 노출할 수 있다.

- [0054] 컴포넌트 선택부(252)는 개발자 인터페이스를 통해 노출된 컴포넌트들 중 하나 이상의 컴포넌트를 선택할 수 있다. 이때, 컴포넌트 선택부(252)는 노출된 컴포넌트들 중 새로운 프로그램 3을 생성하는 데 필요한 컴포넌트 선택을 개발자로부터 입력 받을 수 있다. 여기서, 개발자 인터페이스로는 XML 기반의 동적 개발자 인터페이스가 이용될 수 있다.
- [0055] 일례로, 컴포넌트 1(410)은 게임인 프로그램 1(4100)의 웹 제어(Web Control), 모듈이고, 컴포넌트 2(420)는 프로그램 1(4100)의 화면 크기 조정 모듈이고, 컴포넌트 3(430)은 메신저인 프로그램 2(4200)의 메신저 로그 정보에 대한 모듈이고, 컴포넌트 4(440)는 프로그램 2(4200)의 디자인 모듈인 경우, 컴포넌트 선택부(252)는 개발자 인터페이스를 통해 노출된 컴포넌트 1(410) 내지 컴포넌트 4(440) 중에서 컴포넌트 2(420) 및 컴포넌트 4(440)의 선택을 입력 받을 수 있다.
- [0056] 그러면, 프로그램 생성부(253)는 선택된 컴포넌트 2(420) 및 컴포넌트 4(440)를 결합하여 새로운 프로그램 3을 생성할 수 있다.
- [0057] 도 4는 본 발명의 일실시예에 따라 로딩된 프로그램을 구성하는 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 방법을 설명하기 위해 제공되는 흐름도이다.
- [0058] 먼저, 프로그램 로딩부(240)는 등록된 컴포넌트들을 구성하는 하나 이상의 프로그램을 로딩할 수 있다. 일례로, 프로그램 1(4100) 및 프로그램 2(4200)가 등록된 경우, 프로그램 로딩부(240)는 프로그램 1을 로딩할 수 있다(S510). 이때, 프로그램 1(4100)은 컴포넌트 1(410) 및 컴포넌트 2(420)로 구성되고, 프로그램 2(4200)는 컴포넌트 3(430) 및 컴포넌트 4(440)로 구성될 수 있다. 여기서, 등록된 컴포넌트 1(410) 내지 컴포넌트 4(440)는 커뮤니케이션 버스(300)를 통해 공유될 수 있다.
- [0059] 이어, 인터페이스 노출부(251)는 로딩된 프로그램을 구성하는 컴포넌트들을 개발자 인터페이스를 통해 개발자에게 노출할 수 있다(S520). 일례로, 인터페이스 노출부(251)는 프로그램 1(4100)을 구성하는 컴포넌트 1(410) 및 컴포넌트 2(420)를 개발자 인터페이스를 통해 노출할 수 있다.
- [0060] 그리고, 컴포넌트 선택부(252)는 노출된 컴포넌트들 중에서 하나 이상의 컴포넌트 선택을 입력 받을 수 있다. 일례로, 컴포넌트 선택부(252)는 노출된 컴포넌트 1 및 2(410, 420) 중에서 컴포넌트 2(420)의 선택을 개발자로부터 입력 받을 수 있다.
- [0061] 그러면, 프로그램 생성부(253)는 선택된 컴포넌트를 이용하여 새로운 프로그램을 생성할 수 있다. 일례로, 프로그램 생성부(253)는 선택된 컴포넌트 2(420)를 이용하여 새로운 프로그램 3을 생성할 수 있다(S530). 이때, 복수의 컴포넌트 선택이 입력된 경우, 프로그램 생성부(253)는 선택된 복수의 컴포넌트들을 결합하여 새로운 프로그램 3을 생성할 수 있다.
- [0062] 또한, 프로그램 생성부(253)는 프로그램 3을 생성하는 데 필요한 하나 이상의 신규 컴포넌트를 생성할 수도 있다. 그러면, 프로그램 생성부(253)는 생성한 신규 컴포넌트와 선택된 컴포넌트를 결합하여 새로운 프로그램 3을 생성할 수도 있다.
- [0063] 지금까지 도 4에서는 하나의 프로그램이 로딩된 경우에 개발자 인터페이스를 통해 새로운 프로그램을 생성하는 과정에 대해 설명하였다. 이하에서는, 도 5를 참조하여, 복수의 프로그램이 로딩된 경우에 개발자 인터페이스를 통해 새로운 프로그램을 생성하는 과정에 대해 설명하기로 한다.
- [0064] 도 5는 본 발명의 일실시예에 따라 커뮤니케이션 버스를 통해 공유되는 복수의 컴포넌트들을 이용하여 새로운 프로그램을 생성하는 방법을 설명하기 위해 제공되는 흐름도이다.
- [0065] 도 5에 따르면, 프로그램 로딩부(240)는 등록된 컴포넌트들을 구성하는 복수의 프로그램을 로딩할 수 있다. 일례로, 프로그램 1(4100)이 로딩된 경우, 프로그램 로딩부(240)는 프로그램 1(4100)에 프로그램 2(4200)를 로딩할 수 있다(S610). 즉, 프로그램 로딩부(240)는 프로그램 1(4100)을 메인으로 실행하는 런처(Launcher)에서 프로그램 2(4200)를 부가적으로 실행할 수 있다.
- [0066] 그러면, 인터페이스 노출부(251)는 로딩된 프로그램 1(4100)을 구성하는 컴포넌트와 프로그램 2(4200)를 구성하는 컴포넌트를 개발자 인터페이스를 통해 노출할 수 있다(S620). 여기서, 개발자 인터페이스로는, XML 기반의 동적 개발자 인터페이스가 이용될 수 있다.
- [0067] 일례로, 인터페이스 노출부(251)는 제1 내지 제4 컴포넌트(410~440)를 개발자 인터페이스를 통해 개발자에게 노출할 수 있다. 이때, 노출되는 컴포넌트의 타입은 동적 라이브러리(DLL) 및 실행 프로그램(EXE) 중 어느 하나

일 수 있다.

- [0068] 이어, 컴포넌트 선택부(252)는 노출된 컴포넌트들 중에서 하나 이상의 컴포넌트 선택을 입력 받을 수 있다.
- [0069] 그러면, 프로그램 생성부(252)는 선택된 컴포넌트를 이용하여 새로운 프로그램을 생성할 수 있다(S630). 일례로, 컴포넌트 2 및 컴포넌트 3이 선택된 경우, 프로그램 생성부(252)는 컴포넌트 2 및 3을 결합하여 프로그램 3을 생성할 수 있다.
- [0070] 이때, 프로그램 생성부(252)는 프로그램 3을 생성하는 데 필요한 신규 컴포넌트를 생성할 수도 있다. 그러면, 프로그램 생성부(253)는 생성된 신규 컴포넌트와 선택된 컴포넌트들을 결합하여 새로운 프로그램 3을 생성할 수도 있다. 이에 따라, 개발자가 실행 중인 프로그램에 로딩된 컴포넌트들을 제어하는 개발자 인터페이스를 제공하고, 개발자 인터페이스를 통해 선택된 컴포넌트들을 결합하여 새로운 프로그램을 생성함에 따라 컴포넌트를 재사용할 수 있다.
- [0071] 이상에서 설명한 바와 같이, 프로그램을 구성하는 컴포넌트들은 웹 제어, 화면 크기 조정, 메신저 로그 정보, 디자인에 대한 모듈들로 구성될 수 있을 뿐만 아니라, 배경 음악 제어(Music Control), 화면의 스킨(skin) 변경, 버튼 제어(Button Control), 텍스트 제어(Text Control), 및 탭 제어(Tab Control)에 대한 모듈들로 구성될 수도 있다. 이에 따라, 개발자가 원하는 컴포넌트들을 선택하여 결합함으로써 새로운 프로그램을 쉽고 빠르게 생성할 수 있다.
- [0072] 한편, 본 발명의 일 실시예에 따른 개발자 인터페이스 시스템(200)에서, 등록된 컴포넌트들은 데이터를 직렬화(Serialize), 또는 암호화하여 커뮤니케이션 버스(300)를 통해 컴포넌트 공유부(220)로 전달할 수 있다. 그러면, 컴포넌트 공유부(220)는 직렬화 또는 암호화된 데이터를 다른 컴포넌트로 전달할 수 있다. 이때, 커뮤니케이션 버스(300)를 통해 직렬화 또는 암호화된 데이터를 수신한 컴포넌트는 데이터를 직렬화 해제 또는 복호화할 수 있다.
- [0073] 다른 한편, 이상에서는 신규로 컴포넌트를 등록하고, 커뮤니케이션 버스를 이용하여 등록된 컴포넌트와 기등록된 컴포넌트 간에 데이터를 공유하는 과정에 대해 설명하였으나, 이외에, 본 발명의 일 실시예에 따른 개발자 인터페이스 시스템(200)은 기등록된 컴포넌트 중에서 등록 해제를 요청한 컴포넌트를 해제하는 컴포넌트 등록 해제부를 더 포함할 수도 있다. 그러면, 기등록된 컴포넌트들은 등록이 해제된 컴포넌트와 더 이상 데이터를 공유하지 않을 수 있다.
- [0074] 본 발명의 실시 예에 따른 방법들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다.
- [0075] 이상과 같이 본 발명은 비록 한정된 실시예와 도면에 의해 설명되었으나, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상의 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다.
- [0076] 그러므로, 본 발명의 범위는 설명된 실시예에 국한되어 정해져서는 아니 되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등한 것들에 의해 정해져야 한다.

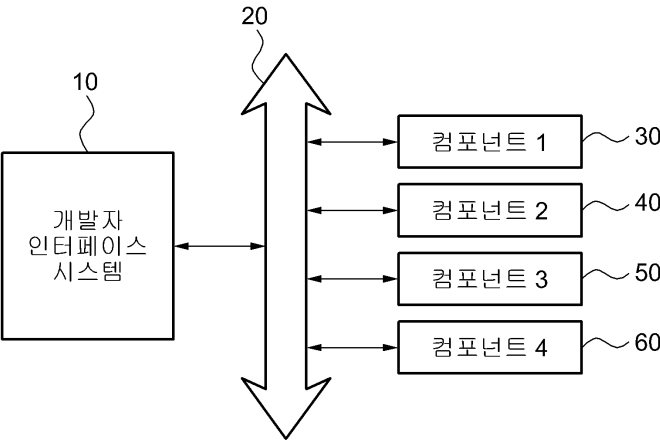
부호의 설명

- [0077] 200: 개발자 인터페이스 시스템
- 210: 컴포넌트 등록부
- 220: 컴포넌트 공유부
- 230: 등록 취소부
- 240: 프로그램 로딩부
- 250: 생성부
- 251: 인터페이스 노출부

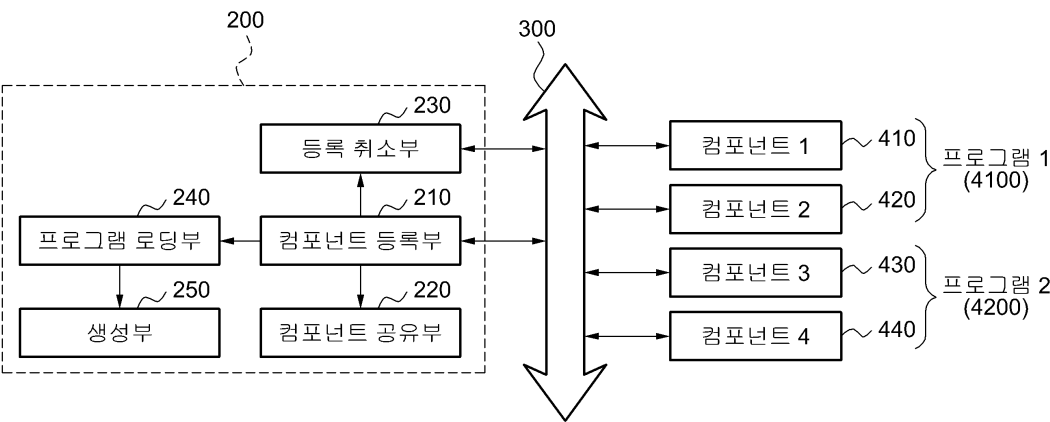
- 252: 컴포넌트 선택부
- 253: 프로그램 생성부
- 300: 커뮤니케이션 버스

도면

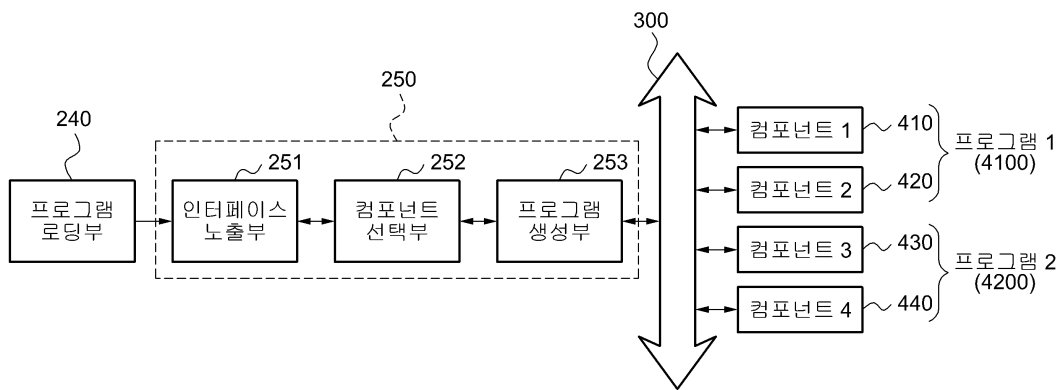
도면1



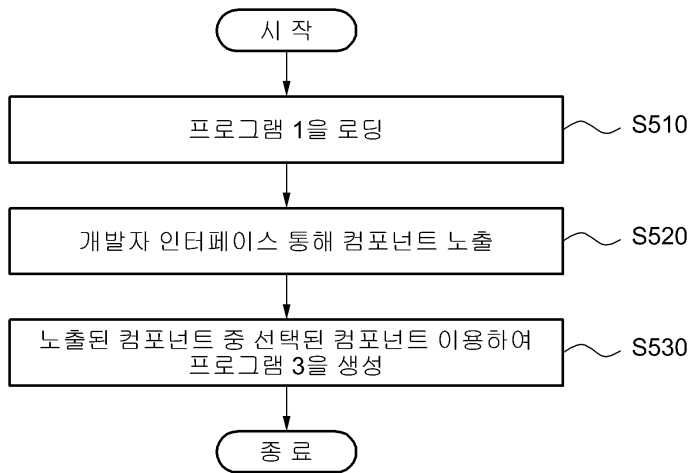
도면2



도면3



도면4



도면5

